Eurasian Journal of Analytical Chemistry
ISSN: 1306-3057 OPEN ACCESS 2018 13 (SP): 339-344

Review of Clinical Decision Support Systems Implementations

Gitanjali Kanakaraj* and Dr.S. Thenmalar

Received 08 November 2018 ▪ Revised: 30 November 2018 ▪ Accepted: 06 December 2018

Abstract: Clinical decision support system (CDSS) is a software, that provides intelligently filtered patient specific clinical guidelines, triggered as computerized alerts and reminders to the physicians. The CDS alerts can be designed as rules using various health information technology codes and computerized guidelines. Doctors can use these alerts to better manage and more accurately diagnose and treat their patients effectively. Numerous models and techniques have been used to enhance CDS systems, and the aim of this paper is to discuss these techniques and models used by them.

Index Terms: Clinical Decision Support Systems, Clinical Decision Support Alerts, FHIR (Fast Healthcare Interoperability Resources).

INTRODUCTION

A clinical decision support system (CDSS) is an expert system used in the field of healthcare to provide assistance to healthcare providers with clinical decision-making tasks. It checks general patient conditions with a knowledge base and then offers suggestions to guide health providers and clinicians. CDS tools include alerts and reminders to physicians and patients, providing document templates triggered by patient conditions, generating order sets for patients etc.

CDSS can be used in assisting with patient-related decision making, determining specific optimal treatment strategies, and helping in drafting general health policies by analyzing and extrapolating the clinical and economic outcomes of different treatment methods when randomized trials are not possible or cannot be trusted.

There are however a lot of problems in the use of CDSS efficiently. Some general problems include alert fatigue and the general mapping of the biological system to a set of machine-understandable rules.

This paper presents a review on CDSS techniques, current and existing methods, and limitations of current methods.

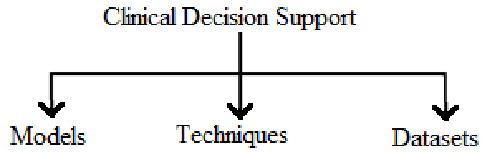
GENERAL CDSS ARCHITECTURE

CDSS generally have three main components. They are Knowledge base, Inference Engine and Communications. The Knowledge base stores information regarding the domain (i.e. the rules and guidelines) that are developed by domain knowledge experts. The rules and guidelines are stored as machine accessible formats and are available to the Inference Engine. These guidelines are sometimes manually developed by domain experts, while at other times, the data needed might be derived using machine learning techniques. The inference engine uses the information given by the Knowledge base and implements the CDS logic. This is usually tailor-made to particular patients. They are used to determine if some action needs to be done or not (for example, alerting a physician to remind a patient to take a particular test). Inference engines have three main instructions they follow; match, select and execute rules. One of the common implementations of inference engines focus on forward chaining, which is trying to work with the logic sequence from the starting condition and coming to the end sequentially. Another implementation is by using Bayesian networks. Since Baye's rule helps to express conditional probability, it can be used to match rules that influence certain conditions if they meet a certain probability. The communications module is used to connect the knowledge base and the inference engine. It is also used to display alerts to the user. It can be in the form of relaying information from the knowledge base to the inference engine and vice versa.

K. Gitanjali Kanakaraj*, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kancheepuram District, Chennai, Tamil Nadu, India. E-mail: gitanjalik_akanakaraj@srmuniv.edu.in

Dr.S. Thenmalar, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kancheepuram District, Chennai, Tamil Nadu, India. E-mail: thenmalar.s@ktr.srmuniv.ac.in

OUTLINE OF SURVEY



CDSS require models for researchers and developers to be able to structure the system. Various techniques are used to improve the inference engine used to generate information and make predictions using raw data. CDSS deals with patient data, developers may use real world data or manually created data for testing purposes.

MODELS

Models are representations of how a system's objects are structured and how they work together as part of a whole. Researchers have developed various models for different diseases and purposes.

Aihua Fan et al. [5] while implementing their CDSS, in the application layer, used the CLP-CPG Model (Constraint Logic Program Model). This model combines clinical practice guideline with constraint logic programming (CLP). The CLP-CPG model was developed due to diagnosis and treatment plans for a patient may be made by combining multiple CPGs which might help to develop a treatment plan for patients with co-morbidities. Tanvir Ahamed et al. [1] developed a model for nursing-specific CDSS. They developed a three-stage approach that builds and assesses a meta-model that is able to manage the requirements of guideline-based clinical nursing-specific CDSS. Since their research is still ongoing, they presented the first two approaches and highlighted the importance of meta-modelling as a tool to find important system-centric information. Sarah E. Rosenbaum et al. [18] developed an Evidence to Decision (EtD) framework that could be used for making well-thought out decisions and recommendations about practice interventions, diagnostic and screening tests, coverage, and health system and public health options. They also created integrated EtD (iEtD) tool that provided the way to build the functionalities of the Evidence based framework.

Researchers used the HL7 FHIR standard (Health Level 7, Fast Healthcare Interoperability Resources standard) to help design their models for their specific uses. Anna M Lackerbauer et al. [13] identified the design requirements necessary for an electronic treatment consent (eConsent) architecture and proposed an architecture that ensured a standardized data model adhering to the FHIR (Fast Healthcare Interoperability Resources) standard. Peng Zhang et al. [24] developed a blockchain-based system called FHIRchain which made use of the FHIR standards which could be used to exchange clinical data in a secure way. Suranga N. Kasthurirathne et al. [12] developed a FHIR module that could exchange clinical data as standard resources. They compared it to the OpenMRS API (Open Source Medical Record System), which was largely application specific, while the FHIR module is standards based API, the second being easier to help exchange clinical data. Crescenzo Diomaiuta et al. [4] identified concepts such as patient, healthcare professional, prescription order and reasons, measurements and simple assertions about a patient. They then mapped them to their associated FHIR resources while developing their module to create clinical documents that helped heathcare clinicians.

Sustainable Medical Applications and Reusable Technologies (SMART) is a platform that creates standards on how Electronic Health Records (EHRs) authenticate and integrate patient data. To keep this constancy, SMART is often used with FHIR standards and applications based and are called SMART on FHIR applications. They are thus SMART-compliant EHR system on top of a FHIR server. Ryan A. Hoffman et al. [8] discussed how the standards used for mortality reporting can be done by mapping FHIR resources to their appropriate standard information used in death certificates and also explored the feasibility of working with the Sustainable Medical Applications and Reusable Technologies (SMART)-on-FHIR. Shyamashree Sinha et al. [20] also used the SMART on FHIR platform for interactivity. They also combined CDSS development and Reusable Technologies (SMART) for CDSS app development by a third party.

Na Hong et al. [9] developed a framework that used the FHIR standards to combine structured and unstructured EHR data into a standard data model.

TECHNIQUES

CDS systems are of two broad types: Knowledge based and non-knowledge based. Knowledge based CDSS have a reference table with all the relevant information. They use traditional AI methods (such as conditional logic) to come to conclusions on the treatment of diseases. Knowledge Based CDSS have three main parts; the knowledge base, the inference engine and the user communication method. The knowledge base contains the various information, the inference engine makes use of the user-developed rules and the patient information to come to a decision. The communication method is the way in which these systems communicate with each other and also the user interface. Knowledge based CDSS make use of decision rule trees, Knowledge based systems need a lot of space to store the various rules and data and the rules are usually derived manually. Non-knowledge based CDSS do not use manually prepared knowledge base and instead derive their information from clinical data, analyzing specific patterns or learning from previous experience by using various machine learning algorithms like Neural Networks and Genetic algorithms. While these algorithms reduce the manual effort it takes to create a knowledge base, these usually focus on a narrow list of symptoms for only a single specific disease while Knowledge based CDSS can be expanded to a larger variety of diseases.

Na Hong et al. [9] developed their framework, that consisted of a FHIR-based, clinical Natural Language Processing (NLP) pipeline and validated it using medication data. They had rule-based system for matching the NLP output types to the similar FHIR elements and also looked into the FHIR elements being a member to the source of the EMR data. Livvi Li Wei Sim et al. [19] used MySQL , HTML, CSS and Bootstrap to develop the web-based interface of the DiabetesDashboard. They used Python language to develop the various CDSS tools, such as the colour-coding, the alert, and the interactive graph module. They did not use any algorithms since their CDSS was a tier 2 model. Henry Odikwa et al. [7] created a hybrid model of CDSS by combining both knowledge and non-knowledge based systems for the diagnosis of diseases by using health care records of prostate cancer and diabetes datasets to train the model. They adopted the min-max method in normalizing the datasets. Multilayer Perceptron (MLP) was used to classify the patients into diseases they suffered from (in this case the system searched and classified for patients suffering from Prostate Cancer and Diabetes) and the genetic algorithm was used in initiating the training weights of the MLP. They used a rule-based reasoning algorithm to build the clinical decision support system. Aihua Fan et al. [5] used the Kerberos and Hadoop (Hbase) tools to build their CDSS, which had three layers, physical, data and application layer. They used CLP-CPG model to find the diagnosis on the co-morbid patients. K.E. Ravikumar et al. [17] used SQL queries, and a simple rule-based approach NLP algorithm in their CDSS and made several enhancements to their previous implementations of a CDSS, making it more accurate. They were able to partially address some programming errors, and completely address some evaluation errors (i.e. clinicians arriving at the wrong decisions). The CDSS they developed was also able to capture the clinical decision through decision logic by altering the implementation based on expert feedback. Sudhir Anakal et al. [2] developed a CDSS for Chronic Obstructive Pulmonary Disease (COPD) by using several machine learning algorithms. They first used Support Vector machine algorithm to model the first assessment of COPD. Multilayer Perceptron Neural Network with three layers using back propagation was used to evaluate the severity of the disease. Their CDSS also enables improvement of treatment outcomes and detects the interaction of drugs and the side effects associated with it by using knowledge base system and prediction techniques. Donald A Szlosek et al. [21] found that there were many CDSSs were not properly evaluated manually. Therefore they evaluated CDSSs automatically by using NLP and machine learning techniques (C-Support Vector Classification (SVC), Decision Tree Classifier, and K-Neighbors Classifier). They compared the results brought in by the three algorithms based on their accurate predictions for judging the occurrence of a mild traumatic brain injury to a manual review and then compared the best results (which were the SVC's results) to the ones measured by the Best Practice Advisories (BPA). Yicheng Jiang et al. [21] developed a three-layer knowledge base model using a multi-symptom Naive Bayes algorithm, the three layers being disease-symptom-property. The specificity of the disease symptoms is weighted by the estimating the probability of contribution of a symptom to diagnose the disease. The system thus iteratively calculates the probability that a patient is suffering from a particular disease using naive Bayes. Yung-Fu Chen et al. [3] designed their CDSS with an integrated genetic algorithm (GA) and support vector machine (SVM). GA was used to select salient features and adjust some SVM parameters (cost and kernel value parameters), while the SVM was used to classify the different categories and calculate the different fitness values. These two algorithms were chosen due to the data being of the imbalanced type. Other algorithms that used such as logistic regression, decision tree, standard neural network, and support vector machine are suited for balanced data. Dao Thi Anh Nguyen et al. [15] developed a CDSS for video head impulse test (vHIT) based on Fuzzy Logic Inference System. The positions of the eye and head movement, calculates the vestibulo-ocular reflex (VOR) gain and uses the fuzzy based inference system to give the results: the normality and the artifact index of the test result. Felix Gräßer et al. [6] used Collaborative Recommender and Demographic-based Recommender algorithms to build their Therapy Decision Support System. Collaborative Recommender algorithm provided better accuracy in determining the individual response to therapy but can only do so for a limited number of cases. Conversely, Demographic-based recommender algorithm gave an overall worse average but could cover the entire database. Thus the therapy decision system was created by combining both the algorithms.

In CDSS, a lot of the information is structured, but the data that is present in the patient notes areas are difficult to extract data from. Thus NLP techniques have been used to extract this data as well. Shervin Malmasi et al. [14] built an open source solution, Canary, by combining the pattern matching and parsing approaches (dependency parsing). Instead of a complete language parser Canary allows users to create a grammar that allows them to model their target information, making it tailor-made for different diseases. Jeffrey Thompson et al. [22] proposed a new algorithm called it the Relevant Word Order Vectorization for structuring free-text. They combined it with a neural network, which could be used to better recognize data from areas where it is traditionally more difficult to extract information.

DATASETS

The existing CDSS make use of either real world data, where the individual markers of the patient had been removed, or false data that mimicked real data and is usually evaluated by clinicians.

Henry Odikwa et al. [7] used the data of 500 prostate cancer patient database and 100 diabetic patients from 2012 to 2015 at Federal Medical Center, Umuahia, and PathConsult Nigeria Limited. This data was then used in the GA and multilayer perceptron algorithms to predict if a patient had either Prostate Cancer or Diabetes. Jeffrey Thompson et al. [22] had a dataset that consisted of tumor pathology reports of women with breast cancer who had treatment at the University of Kansas Medical Center. They had to identify the status of three important breast cancer biomarkers so that they could find if a patient was positive for breast cancer or not. K.E. Ravikumar et al. [17] used data from 21 years and older female patients who visited their clinicians from Employee and Community Health at Mayo Clinic. The CDSS they developed extracted a lot of patient data such as hysterectomy, risk factors, such as cancer, immunodeficiency, and HIV from structured data sources. Lars O. Karlsson et al. [11] conducted a randomized trial involving 43 clinics in Oestergoetland, Sweden to prove that CDS improved the adherence to safety guidelines for anticoagulant therapy in patients with AF. Shervin Malmasi et al. [14] used data from the clinical records of all adult patients treated in primary care practices under the Massachusetts General Hospital and Brigham and Women's Hospital. Four studies were carried out, to check the accuracy of their text extraction. Na Hong et al. [9] used Medication data from Mayo clinic EHR to prove that their framework with FHIR was accurate. Ryan Hoffman et al. [8] used 2012's Multiple Cause-of-Death Mortality Data that contains 2,547,864 deaths to build their Intelligent Mortality Reporting system. Shyamashree Sinha et al. [20] also used Medication data from EHR. They had focused on finding patients who might be at higher risk for opioid overdose or opioid use disorder. Yung-Fu Chen et al. [3] used imbalanced National Health Insurance Research Database (NHIRD) dataset to test their system. Statistical results showed that patient demographics, and corticosteroid-related variables (cumulative dose, mean exposed daily dose, follow-up duration, and exposed duration) were different between patients with and without fractures, and used these parameters to improve their predictive performance. This enabled them to be able to tell if a patient was diagnosed with hip fracture or vertebrate fracture after receiving inhaled corticosteroid therapy.

Some authors like Donald A. Szlosek et al. [21] created dummy datasets which were generated with the help of clinicians and medical students so as not to disturb real world patient data.

IMPLEMENTATIONS OF CDSS

The CDSS have been widely implemented to assist caregivers. There are many open source implementations of CDSS including the old MYCIN system that was supposed to be a diagnostic support system, DXplain which supports 2,400 diagnoses and Isabel, which is the more recent and is a web-based CDSS [23].

Aihua Fan et al [5] designed and implemented a CDSS for co-morbidity monitoring. They developed the architecture, went through the implementation process of building the CDSS and assessed the system accuracy. Dao Thi Anh Nguyen et al [15] presented a CDSS for video head impulse test using two recommendation applications instead of one. Sudhir Anakal et al [2] developed a CDSS for Chronic Obstructive Pulmonary Disease (COPD) that checks for co-morbidities and also has a quit smoking test that informs the patient to quit smoking if they are diagnosed with COPD. Yicheng Jiang et al [10] developed a CDSS based on the three layer model so that more information could be used. Instead of the disease-symptom model, they proposed and implemented the disease-symptom-property model using multi-symptom naive Baye's algorithm.

Several open source CDS systems have been implemented such as OpenCDS which currently also supports the FHIR standards [16].

CONCLUSION

CDSS use different techniques and have modeled on various parameters. It can be seen that a great deal of researchers use the FHIR structure, and many of the systems are non-knowledge based. However, these non- knowledge based systems tend to be in the first stages, or focus on only a single disease. A large number of the systems described in the papers use some NLP techniques to extract data from unstructured sources.

ACKNOWLEDGEMENTS

We are grateful to the Management of SRM Institute of Science and Technology and Dr. B.Amutha, Head, Department of Computer Science and Engineering for guidance and support to carry out this work.

REFERENCES

[1] Ahamed, T., Lederman, R., Bosua, R., Verspoor, K., Buntine, W. and Hart, G. (2016). Towards a Methodology for Nursing-Specific Clinical Decision Support Systems (CDSS). Journal of Decision Systems, 25(sup1), pp.23-34.

[2] Anakal, Sudhir and Sandhya, P. (2017). Clinical decision support system for chronic obstructive pulmonary disease using machine learning techniques. 1-5. 10.1109/ICEECCOT.2017.8284601.

[3] Chen, Y., Lin, C., Wang, K., Rahman, L., Lee, D., Chung, W. and Lin, H. (2018). Design of a Clinical Decision Support System for Fracture Prediction Using Imbalanced Dataset. Journal of Healthcare Engineering, 2018, pp.1-13.

[4] Diomaiuta C., Sicuranza M., Ciampi M. and De Pietro G. (2017). A FHIR-based System for the Generation and Retrieval of Clinical Documents.In Proceedings of the 3rd International Conference on Information and Communication Technologies for Ageing Well and e-Health - Volume 1: ICT4AWE, pages 135-142.

[5] Fan, A., Lin, D. and Tang, Y. (2017). Clinical Decision Support Systems for Comorbidity: Architecture, Algorithms, and Applications. International Journal of Telemedicine and Applications, 2017, pp.1-10.

[6] Gräßer, F., Beckert, S., Küster, D., Schmitt, J., Abraham, S., Malberg, H. and Zaunseder, S. (2017). Therapy Decision Support Based on Recommender System Methods. Journal of Healthcare Engineering, 2017, pp.1-11.

[7] Henry, O., Chidiebere, U. and Hycinth, I. (2017). An Improved Model for Clinical Decision Support System. International Journal of Artificial Intelligence & Applications, 8(6), pp.37-55.

[8] Hoffman, R., Wu, H., Venugopalan, J., Braun, P. and Wang, M. (2018). Intelligent Mortality Reporting With FHIR. IEEE Journal of Biomedical and Health Informatics, 22(5), pp.1583-1588.

[9] Hong, N., Wen, A., Shen, F., Sohn, S., Liu, S., Liu, H., and Jiang, G. (2018). Integrating Structured and Unstructured EHR Data Using an FHIR-based Type System: A Case Study with Medication Data. AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science, 2017, 74-83.

[10] Jiang, Y., Qiu, B., Xu, C. and Li, C. (2017). The Research of Clinical Decision Support System Based on Three-Layer Knowledge Base Model. Journal of Healthcare Engineering, 2017, pp.1-8.

[11] Karlsson, L., Nilsson, S., Bång, M., Nilsson, L., Charitakis, E. and Janzon, M. (2018). A clinical decision support tool for improving adherence to guidelines on anticoagulant therapy in patients with atrial fibrillation at risk of stroke: A cluster-randomized trial in a Swedish primary care setting (the CDS-AF study). PLOS Medicine, 15(3), p.e1002528.

[12] Kasthurirathne SN, Mamlin B, Grieve G, Biondich P.(2015) Towards standardized patient data exchange: integrating a FHIR based API for the open medical record system. Stud Health Technol Inform. 2015;216:932

[13] Lackerbauer, A., Lin, A., Krauss, O., Hearn, J. and Helm, E. (2018). A Model for Implementing an Interoperable Electronic Consent Form for Medical Treatment Using HL7 FHIR. European Journal for Biomedical Informatics, 14(3), pp.37-47.

[14] Malmasi, S., Hosomura, N., Chang, L. S., Brown, C. J., Skentzos, S., & Turchin, A. (2018). Extracting Healthcare Quality Information from Unstructured Data. AMIA Annual Symposium proceedings. AMIA Symposium, 2017, 1243-1252.

[15] Nguyen, D., Won, I., Kim, K. and Kwon, J. (2018). A Development of Clinical Decision Support System for Video Head Impulse Test Based on Fuzzy Inference System. Journal of Sensors, 2018, pp.1-10.

[16] OpenCDS. What Is OpenCDS? OpenCDS Home, www.opencds.org/.

[17] Ravikumar, K., MacLaughlin, K., Scheitel, M., Kessler, M., Wagholikar, K., Liu, H., and Chaudhry, R. (2018). Improving the Accuracy of a Clinical Decision Support System for Cervical Cancer Screening and Surveillance. Applied Clinical Informatics, 09(01), pp.062-071.

[18] Rosenbaum, S., Moberg, J., Glenton, C., Schünemann, H., Lewin, S., Akl, E., Mustafa, R., Morelli, A., Vogel, J., Alonso-Coello, P., Rada, G., Vásquez, J., Parmelli, E., Gülmezoglu, A., Flottorp, S. and Oxman, A. (2018). Developing Evidence to Decision Frameworks and an Interactive Evidence to Decision Tool for Making and Using Decisions and Recommendations in Health Care. *Global Challenges*, 2(9), p.1700081.

[19] Sim, L., Ban, K., Tan, T., Sethi, S. and Loh, T. (2017). Development of a clinical decision support system for diabetes care: A pilot study. *PLOS ONE*, 12(2), p.e0173021.

[20] Sinha, S., Jensen, M., Mullin, S. and Elkin, P. (2017). Safe Opioid Prescripting: A SMART on FHIR Approach to Clinical Decision Support. *Online Journal of Public Health Informatics*, 9(2).

[21] Szlosek, D. and Ferretti, J. (2016). Using Machine Learning and Natural Language Processing Algorithms to Automate the Evaluation of Clinical Decision Support in Electronic Medical Record Systems. *eGEMs (Generating Evidence & Methods to improve patient outcomes)*, 4(3), p.5.

[22] Thompson, J., Hu, J., Mudaranthakam, D.P., Streeter, D., Neums, L., Park, M., Koestler, D.C., Gajewski, B., Mayo, M.S, Relevant Word Order Vectorization for Improved Natural Language Processing in Electronic Healthcare Records Submitted on 6 Dec 2018 , arXiv:1812.02627v1 [cs.CL]

[23] Vardell, E., and Moore, M. (2011). Isabel, a clinical decision support system. Medical Reference Services Quarterly, 30(2), 158-166

[24] Zhang, P., White, J., Schmidt, D., Lenz, G. and Rosenbloom, S. (2018). FHIR Chain: Applying Blockchain to Securely and Scalably Share Clinical Data. Computational and Structural Biotechnology Journal, 16, pp.267-278.